# Infosphere Threats

**by Mr. Timothy L. Thomas**
**Foreign Military Studies Office, Fort Leavenworth, KS.**

**This article appeared in**

***Military Review***

**September-October 1999**

;

ON 3 July 1988, the USS *Vincennes*, located in the Persian Gulf, picked up an Iranian plane on its Aegis system radar. Seven minutes later, the ship's Phalanx gun system blew the plane from the sky. The aircraft turned out to be a civilian airliner and not an F-14 as indicated by the Aegis system. One analysis of the incident noted that "the U.S., and by extension other countries using high-tech weapons, may have become prisoners of a technology so speedy and complex that it forces the fallible humans who run it into snap decisions that can turn into disaster."[1]

This unfortunate incident highlighted some of the emerging problems of the information age: first, the inability of analysts and equipment to visualize the *intent* of electronic images often causes an inaccurate operator "perception-reaction" response; second, a dangerous game of digital roulette results from the inability of software's embedded scenarios to handle all of the anomalies and asymmetric options that develop, by design or otherwise; and third, the impact of electronic input can overwhelm the human dimension of decision making. The analysis suggests the need for a "military software science" to help understand these new phenomena. Such a science would provide a better interpretation and forecast of the scenarios that countries embed in their military software and improve our response posture.

**Implications of the Switch to a Digitized Force**

Force XXI's digitization represents a massive shift away from analog representation data. Analog systems process continuous voltage amplitudes and are costly and specially designed, causing difficulties when sharing information with other systems. Digital systems use rapidly switching "on" or "off" states of binary "1" or "0" as data representations. Digital technology permits a vast decrease in electronic hardware's size and cost, and it allows processing in software rather than hardware. The digital format's resulting flexibility explains our increased reliance on it.

The underlying commonality in all digital signal processing hardware and the ready ability to convert formats and process the information by using software have caused the explosion in information sharing among digital systems. But it is this very ease of trans-mission, extensive processing, changing software and widespread digital data sharing that make intrusion both possible and frightening. If intrusion and corruption succeed, stability disappears and the software's 1s and 0s start falling into unpredictable places, much as the ball that lands unpredictably on a spinning roulette wheel number. If the scenarios embedded in the software are unable to handle unexpected anomalies deliberately introduced by an opponent, stability could suffer.[2] Nations play this game of digital roulette every day with the software in their advanced warning systems, rockets and satellites.

Such a game could result in some serious mishaps or instigate some catastrophic chain reactions of events. For example, what would happen if one side could project false radar blips on a Joint Surveillance Target Attack Radar System (JSTARS) in a manner so realistic, extensive and threatening that a potential opponent expends an arsenal of cruise and other precision-guided missiles on the illusory threat? Could it result in command and control decisions that put nuclear forces in a ready-to-launch status once other assets are exhausted? The world could be thrust on the brink of a holocaust by some false images on a computer display. There are no guarantees that all cultures and nations will include "fail safe" rules in their software to guard against such an accidental launch.

A programmer writes code to fulfill a task but as Ellen Ullman noted, "it is a task as a human sees it: full of unexpressed knowledge, implicit associations, allusions to allusions. Its coherence comes from knowledge structures deep in the body, from experience, memory."[3] Human knowledge mechanisms, as they relate to culture, language and the means of expression, are quite complex. Someone should study the relationship between culture and programming in a systemic way because there is no concrete understanding of the impact of culture on programming. On the one hand, it is reasonable to suggest that if different cultures think differently, there is no reason why this practice might not affect the way they program. On the other hand, there is the view that traditional culture makes little difference in the act of programming. Computers may, for example, be creating a horizontal culture, like rock music and McDonalds, which obliterates traditional national boundaries. Perhaps programmers in Calcutta, while on the job, live and program almost exactly as do programmers in Boston or Silicon Valley.[4]

Retired Air Force Colonel Richard Szafranski, writing on "Neocortical Warfare," *Military Review*, November 1994, noted that F.S.C. Northrop talked about the impact of culture on the brain in 1946, before the development of computers. Northrup's interpretation would fit Ullman's first viewpoint because Szafranski, paraphrasing Northrup, remarked that "Culture conditions some of the operations of the left brain. Specifically, atmospheric and linear perspective in classical Western art and the syntax of romance languages both work together to channel cognition in ways that are different from the ways that the undifferentiated aesthetic continuum of Eastern art and the syntax of the Asian word-
picture or ideogram condition the thinking of those in the East."[5] However, perhaps more pertinent, any difference in programming will be swamped by doctrinal differences in how cultures develop and interpret computer displays, what they design the system to provide, and so

forth. Cultures will specify tasks differently to solve problems, such as the manner in which heads-up displays were developed for the helmets of Russian and US fighter pilots.

Two Russian analysts who studied digitized-age implications added other characteristics of the digital context, noting that:

- Collection is becoming more closely linked with data analysis and processing, and the entire effort is much more integrated than before.
- Human involvement is decreasing, especially in the collection and processing phases.
- Just as virtual reality is blurring the geopolitical boundaries of information space, it also obscures the enemy image—is it a national or transnational threat?
- Distance is not as important as time in making and implementing decisions.
- Software may become hostile to mankind if it spawns systems of growing complexity or self-
  destructs.[6]

**Understanding "Information-Based Intent"**

Intent is an amorphous concept defined as a purpose or goal—why one performs an act. Intent originates in an individual today just as it always has. In the past, analysts measured intent by observing a country mobilize resources, move tanks to the front and deploy into battle formations. Discerning the intent of electrons, their purpose, goal or what act they are performing is another matter. Take, for instance, the difficulty in exposing the intent of electrons sent from a private computer somewhere in the world and rerouted through several intermediate countries. Where did the electrons originate? Who initiated the attack? What is the goal or purpose of the electrons, and what are they doing?

The soldier-operator (S-O) behind a computer monitor or radar screen is usually the front-line defense in the battle to detect and understand electronic intent. The S-O works in and relies on virtual space for his contextual understanding. This workspace is where the tension between space-based information capabilities and intent escalates when commanders and operators face uncertainty yet pressure to react. A Navy captain who "hesitates too long while trying to identify conclusively that radar-screen blip" could lose his ship and the lives of all those aboard.[7] Pressure to react rather than think produces hair-trigger rules of engagement (ROE) for naval forces in the Persian Gulf, requiring "some convincing indication [an electronic image?] that a ship or plane is approaching with hostile intent" to ask headquarters for permission to shoot.[8]

This leads to the frightening conclusion that the tactical operator/strategic decision maker works in a "perception-reaction" chain of events in the information technology (IT) age, a phenomenon that is even more dangerous than perception management. Perception reaction is the knee-jerk impulse that operators/decision makers feel when presented with images and warnings of an imminent attack, as happened in the case of the *Vincennes*. Perception reaction in IT can be understood as "actions based on observations of electronic images that elicit survival or other immediate response actions. Whether these images are real or artificial matters not, as both types will influence emotions, motives or the objective reasoning of individuals or systems."[9]

In the past, when opponents seemed aggressive, planners had time to calculate opposing forces where the unit focus was critical—battalion versus battalion, tanks versus tanks. While correlating opposing electronic forces is possible, correlating opposing electrons is more difficult. The units of measure for such comparisons are simply unknown. Now is there a way to measure their intent and focus—is it at the unit or strategic level?

Electronic intent is easily masked, as skilled computer programmers now demonstrate. For example, screensavers with a soothing appearance could launch viruses designed and destroy the system. Responding to someone's destructive electronic intent is an imprecise science, because the responder must decide how and where to respond, how much is enough and what the originator's intent was in the first place. Such a response is not nearly as clear as reacting to a tank that shoots at you.

**Software's Cultural and Embedded Dimensions**

During the Cold War, Soviet and American analysts studied each other's military establishments in great detail, focusing on examining each other's capabilities, intentions and decision-making processes to expose hostile activity and defeat it quickly. These processes represented a unique combination of the specific linguistic, environmental, historical, philosophical and cultural elements in each country. Unfortunately, analysts often failed to account for these complexities in a nation's
strategic culture. Instead, they resorted to mirror-imaging capabilities and intentions through their own prism of reality.[10] The resulting forecasts, tainted with prejudices and expectations, led to an analysis lacking insight and context. These factors are equally important today when studying the software packages that drive a country's military logic and rationale. They must not be ignored. For example, many factors may direct programmers—a culture's military science and theory, budget restraints, discoveries in hardware and software capabilities, local brain power's ability to create mathematical code, the existing technological infra-
structure (how many layers deep can an analysis or development proceed) and even the rationale of procurement and threat analysis. One programmer noted that the process "has remained maddeningly undefinable, some mix of mathematics, sculpting, scrupulous accounting and wily, ingenious plumbing."[11]

Not all nations may write computer code the same way, nor might they share an international language of logic and rationale. It is affected by many other factors because each country may develop unique scenarios or programmed responses. This may make weapons and associated systems respond differently than planned when confronted by various culturally driven scenarios. We play as we practice, and if there actually are culturally driven variations of computer software, we will have trouble responding properly in training and operations.[12]

In addition, what if our software has been secretly corrupted or damaged, or if there are anomalies or asymmetric scenarios that trigger unintended, incoherent or disjointed software responses? This high-stakes game of digital roulette is further complicated by the fact that some software for Force XXI weapons was developed by hundreds of companies and programmers. As a result, problems may develop from overlapping or contradictory code. As one scientist working closely with antiballistic missile software noted, "Already we are at the point where it is difficult

to understand how software works. Codes are so large, and made up of so many components that have been developed by so many different people, that getting a fail-safe piece of software that is 100 percent understood is virtually impossible. The cost of code validation and verification can easily exceed the cost of writing the damn thing to begin with."[13]

Reprogramming a computer to react to new situations may be nearly impossible in the short term due to human ignorance of the algorithms required for the processes involved and increasing complexities of the software. One answer to the problem of anomalies has been offered by Peircean Semeiotics, the science of self-controlled, deliberate reasoning while focused on problem solving. This body of thought is currently under review by the Pentagon and offers the opportunity to construct a system to deal with and react to a new situation in a nonprogrammed manner. Not all artificial intelligence systems developed to date fit this criterion.[14]

These software features demonstrate the growing complexity of working in virtual space. More important, this puts a new spin on the old saying "if there is the perception of a problem, then we have a problem . . . even if the problem is simply the perception." Will software become a new branch of service as a result? And will the new "enemy image" manifested as software-generated blips and pixels so dehumanize adversaries into images that the use of force will be more likely?

**The Human Dimension**

The increasing reliance on a virtual image of reality complicates decision making. Often, the analysts' training, expectations and frame of reference leave them unprepared to interpret virtual space. From nation to nation, analysts' vision of virtual space varies dependent not only on the software and analytical tools available but also how the cultural and religious philosophies shape their view of reality. Unaccustomed to studying a virtual enemy image, analysts struggle to interpret the virtual images they see or hear within their own context.

The interpretation of electronic images' intent or the detection of electronic waves depends on the picture of virtual reality a specific analyst has acquired. At the same time, the decision-making focus has shifted some from planner/analyst/decision maker to the direction of operator/analyst/decision maker. Greater reliance on the operator's interpretation implies that the tactical operators, not strategic leaders, might make many vital and critical decisions about target engagements.

Another problem of the human dimension is that our reliance on computers is eroding our manual skills to analyze and understand the phenomenon itself. What if we have to abandon our software altogether and rely only on manual operator processes? This latter point appears to be very sensitive since "Perhaps nowhere is our vulnerability to asymmetric technologies greater than in our relentless pursuit of information superiority. Our vulnerability lies in the realization that the more proficient we become at collecting, processing, displaying and disseminating relevant, accurate information to aid decision makers, the more dependent we become on that capability and therefore the more lucrative a target."[15]

The danger is real since "by their very nature as automatons, computer systems have no inherent ability to recognize their own limitations. When applied in inappropriate circumstances, they will

produce answers that may be `logical' but quite incorrect. The entire process, from concept through design, testing and doctrine development, must include a recognition of this inherent problem."[16]

Most commanders still have trouble taking that leap of faith that puts the world's fate in the hands of automatic code generation, hesitancy that may help preserve some manual skills. Yet software forces commanders and decision makers to rely on program code as the new "maestro" of warfare.[17]

Finally, future analysts must focus their attention on the heart of the information weapon, the algorithm. They must become adept in the computer graphics of potential enemies, understand their computer logic as well as the logic of their military art and become familiar with the techniques and contextual factors within which the logic is developed. Computers alone cannot analyze this human aspect—people must collect and sort data, then digest and reproduce it into some comprehensible form. The "network warrior analyst" will be one of the most important future US soldiers.

**The Need for a Military Software Science**

The foregoing discussion has described the concept of electronic intent, the problems associated with so much reliance on software programming and the role of the human dimension in the problem. Such developments suggest the potential for a new branch of theory called *military software science*. This scientific activity would serve as a clearing house to analyze various types of military software logic. Military software science would require very specialized analysts from the software engineering, culture and military history and art domains. They would form a military software science directorate within the Department of Defense. Software's importance cannot be overstated. It integrates, coordinates, allocates and synchronizes our forces, enhancing command and control logic and supervision, while the overseer of the battlefield analysis process remains the commander who applies the principles of war and military strategy.

This discussion has highlighted emerging threats in the "hot spot" known as the infosphere, focusing on some factors that a digitized force will encounter. It is tentative thinking and invites much further exploration.

The argument has been presented that nations seldom consider the cultural aspects that drive another nation's computer software programming mechanism. Further, we have no idea how to measure electronic intent nor a way to trust computer output that, because of its processing speed, detects threats and offers options quicker than humans can comprehend. Nor can we predict what culturally based software might offer as a response mechanism to a perception-reaction scenario. The development of a military software science directorate may help address these issues.

Many countries have already endured a cultural invasion of sorts. Programmers in various countries wrote many of the first programs and perhaps introduced a cultural bias for the way future programs are written that has become invisible to follow-on generations. Militaries

worldwide may enter their own cultural biases via software programming, especially in the realm of military art or the principles of war, which vary from nation to nation.

The issues of determining the intent of electrons and unanticipated software responses should receive special consideration this year in light of the Y2K problem. For example, what happens if radar screens of countries possessing nuclear weapons go blank during the Y2K changeover? Would we be able to distinguish malfunctioning software from a computer attack? Will we someday develop software control theories like today's arms control theories? The first question becomes particularly alarming considering that, for example, Russian military doctrine anticipates an attack on its military information system via a virus or electromagnetic attack before a nuclear strike against Moscow.[18] The United States and Russia are trying to develop an agreement that would put experts in each other's nuclear command centers to prevent any miscalculations if systems go haywire.[19]

To deal with such uncertainties, it appears highly advisable to construct an "information hotline" between countries, similar to our nuclear hotlines, to preclude potential horrific misunderstandings in this sensitive area. Perhaps it would be best to collocate these lines and interests and hold one hostage to the other. Such a hotline would require preliminary work to agree on terminology, concepts and theory, along with discussion on capabilities and intent, and would reduce tension and encourage discussion.

Strategic thinkers should discuss the idea of culturally driven software, the need for a military software science and the consequences of digital roulette and the perception-reaction problem. The central question is whether technology may be pushing the fallible humans who operate it beyond their ability to make wise judgments instantly on the basis of what, with even the most sophisticated systems, will often be ambiguous information. This question applies not only in the Persian Gulf, but wherever there are fingers on buttons that can launch deadly weapons.[20]

Much more time must be spent on these issues. The stability of data in a private citizen's personal computer is one thing, quite another in computers for launching missiles or the minds of decision makers. ***MR***

*The views expressed in this article are those of the author and do not purport to reflect the position of the Department of the Army; the Department of Defense or any other government office or agency.—Editor*

---

1.George C. Church, "High Tech Horror," *Time*, 18 July 1988, 14.

2.Author's discussion with defense scientist Jay Willis.

3.Ellen Ullman, *Wired*, April 1999, 128.

4.Discussion with Ullman via E-mail, 6 April 1999.

5.*In Athena's Camp*, ed. John Arquilla and David Ronfeldt (Rand, 1988), 401.

6.Yuriy Baturin and Sergey Modestov, "Intelligence in Virtual Reality," *Vooruzheniye Politika Konversiya*, No 1-2, 1998, 53-56.

7.Church, 17.

8.Ibid.

9.Author's definition.

10.For a Western discussion of Russian and US capabilities and intents during the Cold War period, see Ken Booth, *Strategy and Ethnocentrism* (New York: Holmes and Meier, 1979); and Fritz Ermarth, "Contrasts in American and Soviet Strategic Thought," *International Security*, Fall 1978, 138-155. Ermarth underlines the values and methods involved in the strategic culture of the arms race. For a book that offers both Russian and US views of strategic culture, see Carl G. Jacobsen, ed. *Strategic Power: USA/USSR* (Macmillan Press, London, 1990).

11.Ullman, Ibid.

12.James Schneider, School for Advanced Military Studies, Fort Leavenworth, KS, adds: "Such a reliance places increased pressure on $C^4I$ and indicates that *software* is replacing *staffware* (or normal staff procedures). The intelligence fusion process is increasingly being taken over by the software (recall the Vincennes incident was virtually a software/fusion problem)." Software, unlike a machine, will continue to operate without indicating that it broke until it is too late, and one must always be looking for software-generated information garbage.

13.Author's conversation with Willis.

14.Discussion with Ed Nozawa of Lockheed, 11 February 1999.

15.Jay M. Garner, "Asymmetric Niche Warfare," *Phalanx*, March 1997, 1. Asymmetric warfare (fighting in cities, use of chemical or biological warfare, domestic terrorism to fix the National Guard, etc.) will leverage focused technologies to adversely affect our employment capabilities, 2.

16.David S. Alberts, "The Unintended Consequences of Information Age Technologies," National Defense University, Washington, DC, 39.

17.Alberts, 40.

18.Colum Lynch, "Y2K Bug Worries U.S., Russia," *Boston Globe*, 12 December 1998, as downloaded from Johnson's list on 12 December.

19.Ibid.

20.Church, 17.